

本サービスにおける著作権および一切の権利はアイティメディア株式会社またはその情報提供者に帰属します。また、本サービスの出力結果を無断で複写・複製・転載・転用・頒布等を行うことは、法律で認められた場合を除き禁じます。

江端さんのDIY奮闘記 EtherCATでホームセキュリティシステムを作る(6):

ご主人様とメイドは、通信の設計図を数秒で書く

<http://eetimes.jp/ee/articles/1512/28/news016.html>

さて、今回はいよいよステッピングモーターを回す方法をご紹介します。その上で、マスター(ご主人様)とスレーブ(メイド)が何をどうしているのかを解説しましょう。実は、ご主人様とメイドは、制御系エンジニアが何週間もかけて書くスレーブのメモリマップ設計仕様書を、わずか数秒程度ですると書き上げてしまうのです。

2015年12月28日 09時30分 更新

[江端智一, EE Times Japan]



FA(ファクトリオートメーション)を支える「EtherCAT」。この超高度なネットワークを、無謀にも個人の“ホームセキュリティシステム”向けに応用するプロジェクトに挑みます……!! ⇒「[江端さんのDIY奮闘記 EtherCATでホームセキュリティシステムを作る](#)」連載一覧

活版印刷(かっぱんいんさつ)、写真、インターネット。

これらの人類のパラダイムシフトをけん引してきたメディアのコアコンテンツが、全て「エロ」だったという話は、あまりにも有名です。

実際、私は、インターネットが「エロ」コンテンツによって — いや、正確に言うと「エロ」コンテンツだけで発展してきた経緯を、目の当たりにしてきた当事者です。

私が高校生のころ、PCなどという機械を操る奴は、一握りの技術エリート — 「オタク」とも言われていましたが — だけでした。電話をつないでPCで通信をするためには、手作業でドライバを組み込む(下手すると「自作する」)などという、超絶な高等技術が必要でした。

そのような情報系のエンジニアですら手を上げてしまうような困難な技術に、果敢に挑戦したのは、理系、文系の枠組みを超えた、日本中の成人男性たちでした。

— 海外のエロ写真を見たい

この純粋な「エロ」への思いは、わが国のネットワーク技術を、エンドユーザーのレベルに至るまで、高度なレベルに引き上げました(今はひっくり返ってしまっているようですが([参考](#)))。

□

IoT(モノのインターネット)にパラダイムシフトを与え得る「エロ」とは、何であろうか。

私は、ずっと考え続けています。

私は、IoTを真に発展させるものは、「ビッグデータ」でも「人工知能」でもなく、「エロ」であると確信しています。しかし、現時点において、そのような観点から論じられている文献、論文、講演などは存在していないようです。

私は、内閣府、経済産業省あたりが、「国家戦略としての『エロ』と『IoT』」を真面目に検討し始めたら、わが国のIoT技術やサービスは、世界的に突出したレベルに至るのではないかと期待しています。

では、なぜ、内閣府、経済産業省、主導なのか？

決まっています。

社内でそんな提案したら、「私のクビ」が危ないからです。

ステッピングモーターを回してみる

こんにちは。江端智一です。

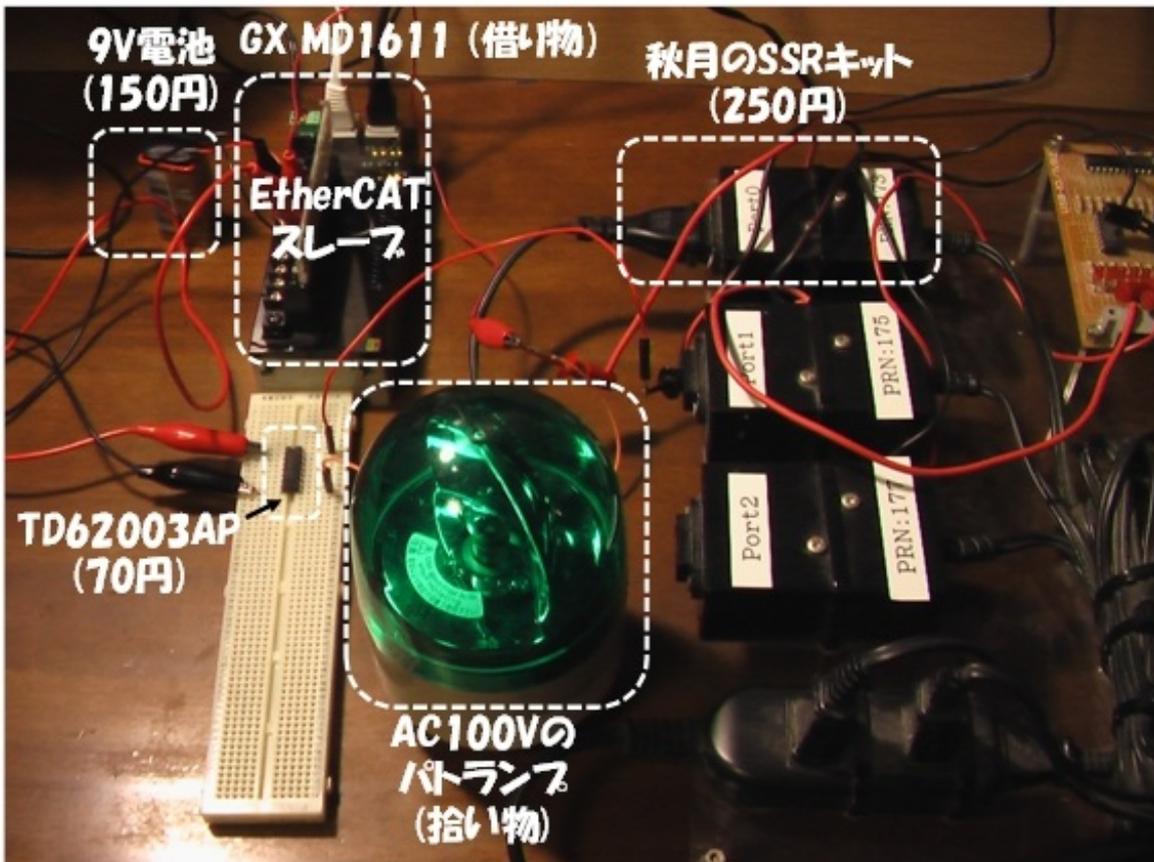
[前回の最後のページ](#)に、「EtherCATスレーブでステッピングモーターを回すことに成功しました!」と記載しました。

そこで、「読者の皆さんにも、実際にステッピングモーターを回してもらおう」ことを目的として、前後半2回に分けてお話ししたいと考えています。

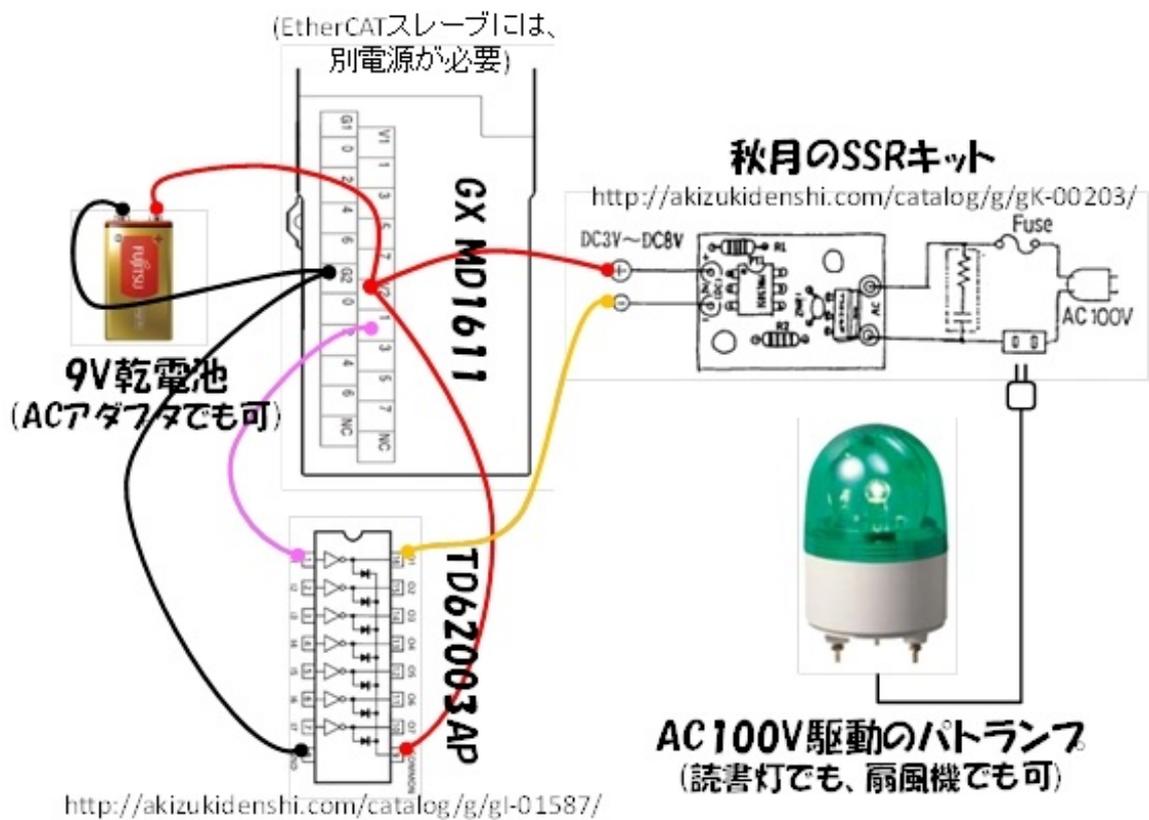
まずは、実際に動いているところをご覧ください。

パトロールランプをオン/オフする様子がこちらです。

この構成図は下記の通りです。



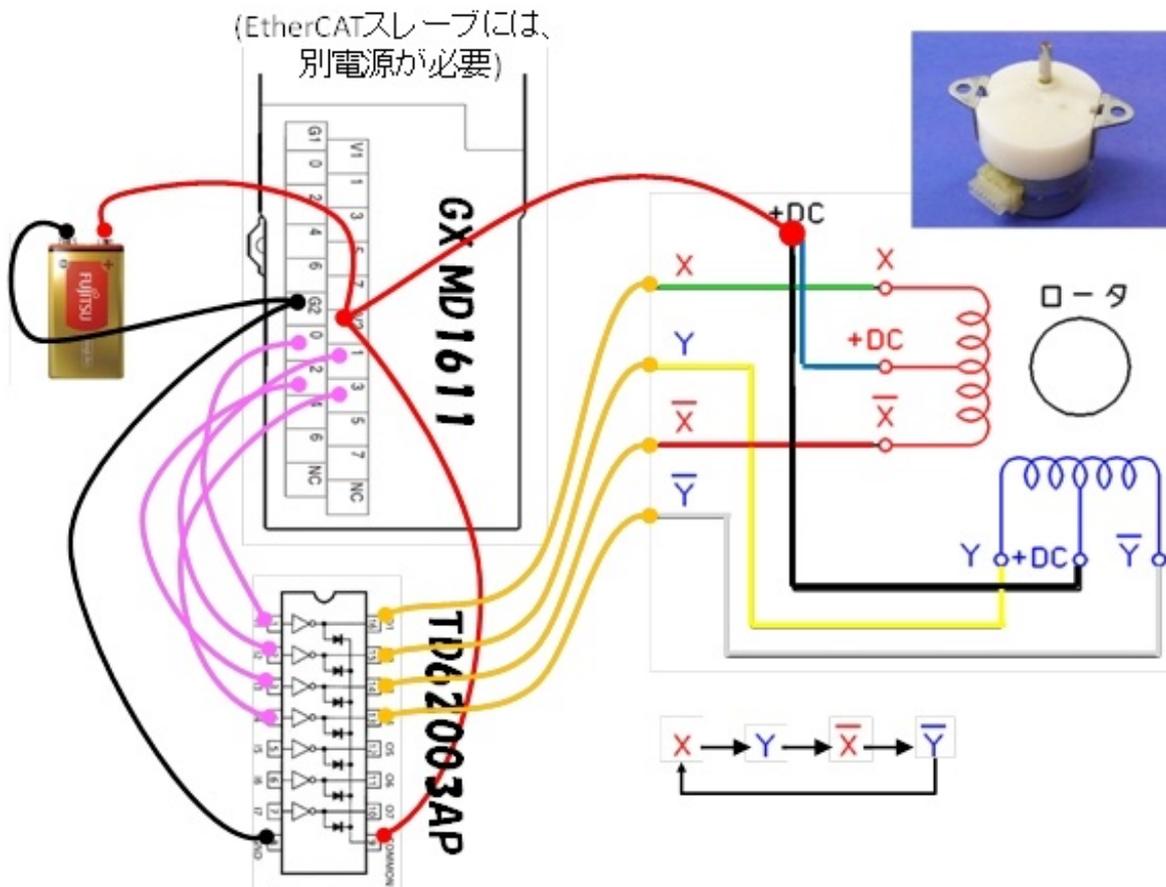
そして、回路図は以下のようになっています。



超低速モードと通常モードで動かす

次の2つの動画は、ステッピングモーター（超低速）と、通常回転モードのものを撮影した様子です。

回路図はこちらになります。



さて、しょっぱなから、写真や動画や、回路図などで戸惑われているかもしれませんが、つまるところ、ここでご紹介しているランプやらステッピングモーターやは、全てEtherCATスレーブのDO（デジタル出力）、つまり信号電圧だけを使って動かしています。

ところで――。

この連載が始まってから（2015年5月）、モーターを回すまで、なぜこんなに時間がかかったと思いますか？ それは、EtherCATを使った実装例が、全くネット上に存在しなかったからなのです。

そもそも、EtherCATのスレーブのハードウェアの仕様がいまひとつよく分からない。

マニュアルでは、モーターを駆動するだけの電気を供給できる用にも読めるけど、接続してみるとピクリとも動かない。規格ギリギリまで電圧を上げた時には、スレーブやモーターから白い煙が上がったこともあります。

ステッピングモーターの勉強も一から始めました。

まず、ステッピングモーターの[キット](#)を購入して組み立てて、実際に「手動」で[モーター](#)も動かしてみました([参考動画](#))。

その後、このキットを分解し、モーターだけを取り出して、EtherCATのスレーブに直結してみたのですが、やっぱり動きません。

そこで、「スレーブは単なるON/OFF信号装置だ」と割り切って、別途にリレーやモーターを駆動するドライバ回路の作成を決意しました……が、——これが、超絶に面倒くさい。

老眼がすごい勢いで進行しているわが身にあって、小さな電子部品を、ブレッドボードに載せたり、はんだ付けしたりとか、ちょっとした拷問です(もうすぐ、あなたにも分かるようになる)。

そんな時、たまたまネットで見つけたのが、「TD62003AP」というICでした。面倒な回路を作る必要なく、直接モーターをドライブできるという、こんなすごいトランジスタアレイICが1個70円。定時で会社を飛び出して、町田のパーツ屋さんでこれを2個購入してきました。

鼻歌まじりにスキップしながら帰る途中、パーツ屋さんの近くにあった、質屋のディスプレイに陳列されたブランドバックの値札が目に入りました。

2万円。

「信号電圧をドライブすることもできない、かばんごときが!」と、憤りを感じたのを覚えています。

閑話休題。

ステッピングモーターを動かす準備

さて、今回の前半では、

- ステッピングモーターの駆動回路とDI/DOのEtherCATスレーブ(この連載では「メイド」との接続方法(前記で説明済み)
- Windows7の上で動くEtherCATマスタSOEM(Simple Open EtherCAT Manager=この連載では「ご主人様」)を使って、メイドとご主人様が、ステッピングモーターを動かすために、どんな取り決めをしているか

を、具体的に説明します*)。

*)実際に、モーターが動くようになるのは次回になりますが、「この年末年始の休暇中に動かしたいんだ!」というエンジニアの方がいらっしゃったら、その熱い思いに応えますので、江端に連絡ください。

ステッピングモーターを回すための準備は、以下の通りです。

【Step.1】

DI/DO (DOだけでも可)のEtherCATスレーブを、どっかからかっばらって来てください。私が稼働を確認しているのは、[このページに掲載しているスレーブ](#)です(ステッピングモーター、TD62003AP、プローブなどは、来月までに手に入れば間に合います)。

【Step.2】

本連載の第4回「ご主人様とメイドはテレパシー通信をしている?」の[こちらの部分](#)を読んで、SOEMのデバッグ&トレース環境を作っておいてください。

では、slaveinfoのプロジェクトを「スタートアッププロジェクトに設定」して、実行をしてみてください。以下のように、呪文のような情報が出てくるはずですよ。

slaveinfo.exeの表示(前半) →接続されている全スレーブのまとめ的信息

EtherCAT用に使っているパソコンのEtherNIC

コンフィギュアードアドレス

デジタル入力:16bit, デジタル出力:16bit

パソコンのこのメモリアドレスに書き込むと、デジタルIOに出力できる

パソコンのこのメモリアドレスを読み込むと、デジタルIOの内容を読みめる

「テレパシー通信」(江端命名)用のパソコンのメモリ(開始)アドレス

SyncManager2と3のスタートアドレス(後述)

(クリックで拡大)

これが、EtherCATマスタ(ご主人様)とスレーブ(メイド)が1000回くらいの情報交換をして、私たち人間の手を一切借りることなく自分たちだけで作り上げた、EtherCAT(お屋敷)の設計図です。

ここで重要なのは、本連載の[第4回](#)と[第5回](#)で説明した「テレパシー通信(江端命名)」用のご主人様(SOEM)のメモリアドレスが決まっていることです。

「テレパシー通信(江端命名)」とは、ご主人様であるマスタ(のPC)のメモリに情報を読み書きするだけで、それが、全部のメイドであるスレーブをコントロールすることです。

ここでは、そのご主人様のメモリの先頭アドレスが「IOMap:01257D80」(1)と決まっている

ことが分かります。

また、スレーブ1の出力デジタルI/O用のアドレスが「Outputs 01257D80」(2)で、スレーブ1の入力デジタルI/O用のアドレスが「Inputs 01257D82」(3)となっているのも分かります。

要するに、ご主人様であるマスタ(PC)の、このアドレスの値を適当にイジるだけで、メイドであるスレーブを意のままに動かすことができるわけです。

『“01257D80”のような、ベタベタなアドレスのメモリをどうやって使うんだ?』と思われるかもしれませんが、心配はいりません。

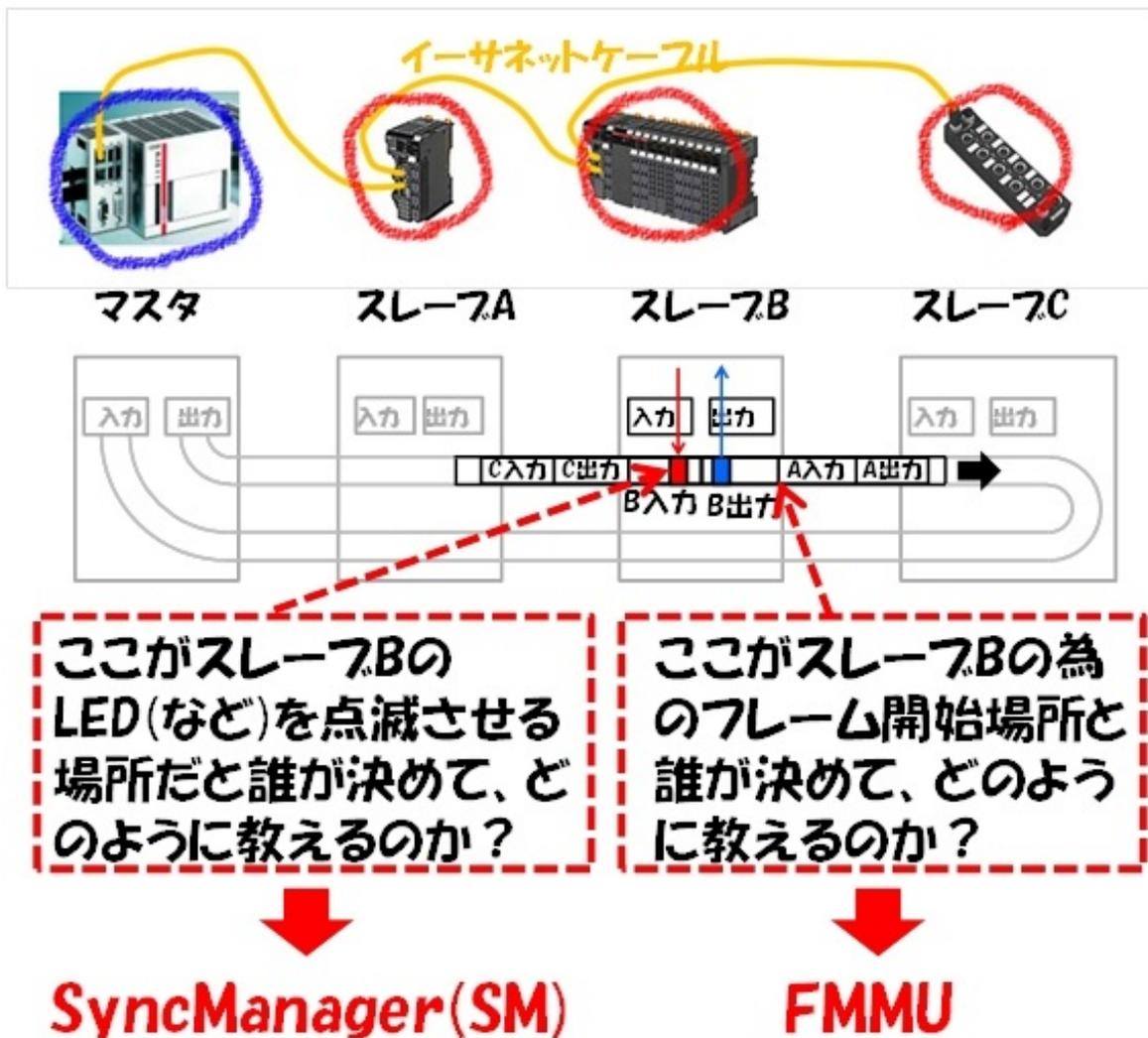
SOEMのソースコード中には、“char IOmap[4096]”という配列が定義されていて、その先頭アドレスとマッピングされているからです。

つまり、今回のケースでは、マスタのソースコードの中のIOmap[0] (“01257D80”) とIOmap[1] (“01257D81”) が出力用変数、IOmap[2] (“01257D82”) とIOmap[3] (“01257D83”) が入力用変数として使えることとなります。

ご主人様(マスタ)は、この“char IOmap[4096]”の変数を使えばいいでしょう。しかしメイド(スレーブ)の方は、自力でEthernetフレームから、自分宛の情報を取り出さなければなりません。

EtherCATの2大技巧とは何か

これを解決する、EtherCATの2大技巧、FMMU (Fieldbus Memory Management Unit) と、SyncManagerを説明します(これが理解できれば、EtherCATの学習はほぼ完了と言っても良いと思います)。



では、最初にFMMUについて説明しましょう。

slaveinfo.exeの表示(後半)
→ 一番目のスレーブ情報の詳細

EtherCATスレーブの製品名

スレーブ間の時間同期の方法(後述)

コンフィギュアードアドレス

SyncManager(メールボックス通信用)(後述)

SyncManager(プロセスデータ通信用)(後述)

FMMUの情報(後述)

```

Slave:1
Name:GX-M01611
Output size: 16bits
Input size: 16bits
State: 4
Delay: 0[ms]
Has DC: 1
DCParentport:0
Activeports:1,0,0,0
Configured address: 1001
Man: 00000083 ID: 00000032 Rev: 00010000
SM0 A:1800 L: 512 F:00010026 Type:1
SM1 A:1c00 L: 512 F:00010022 Type:2 (2)
SM2 A:1100 L: 2 F:00010064 Type:3
SM3 A:1200 L: 2 F:00010020 Type:4
FMMU Ls:00000000 LI: 2 Lsb:0 Leb:7 Ps:1100 Psb:0 Ty:02 Act:01 (1)
FMMU Ls:00000002 LI: 2 Lsb:0 Leb:7 Ps:1200 Psb:0 Ty:01 Act:01
FMMUfunc 0:1 1:0 2:0 3:0
MRX length wr: 512 rd: 512 MRX protocols : 04
CoE details: 23 FoE details: 00 EoE details: 00 SoE details: 00
Ebus current: 0[mA]
only LRD/LWR:0
End slaveinfo, close socket
End program

```

デジタル入力:16bit, デジタル出力:16bit

ベンダ番号、製品番号、バージョン番号

メールボックスのサイズ等(後述)

スレーブには、8bit/8bitのポートしか実装されていないけどね

(クリックで拡大)

FMMUとは、「Ethernetフレームのどこに何の情報が書き込まれているか」を示す、地図情報のようなものです。

```
FMMU0 Ls:00000000 LI: 2 Lsb:0 Leb:7 Ps:1100 Psb:0 Ty:02 Act:01  
FMMU1 Ls:00000006 LI: 2 Lsb:0 Leb:7 Ps:1200 Psb:0 Ty:01 Act:01
```

(1)の部分のここは、実際のプロセス通信を開始する前に、Ethernetフレームのどこに何の情報を搭載するかを、ご主人様(SOEM)とメイド(Slave1:コンフィギュアードアドレス1001)が、事前に話し合っただけ決めたものです。

WireSharkでキャプチャーしたEthernetフレームの図と、併せて表記してみましょう。

EtherCATフレームの0バイトがスタートだよ!

長さは2バイトだよ!

スタートは0bit目、エンドは7bit目だよ!

出力用だよ!

FMMU0 Ls:00000000 LI: 2 Lsb:0 Leb:7 Ps:1100 Psb:0 Ty:02 Act:01

Cmd: 12 (Logical Read) Index: 0xb Log Addr: 0x00000000 Length: 12 (0xc) - No Roundtrip - Last Sub Command Interruption: 0x0000

Data: 0c00000c000000000000

working Ctrl:

0000	ff ff ff ff ff ff 01	スレーブA	スレーブB	スレーブC	スレーブA	スレーブB
0010	0c 0b 00 00 00 00 00	の出力値	の出力値	の出力値	の入力値	の入力値
0020	00 00 00 00 00 00 00					

入力用だよ!

FMMU1 Ls:00000006 LI: 2 Lsb:0 Leb:7 Ps:1200 Psb:0 Ty:01 Act:01

入力	出力	入力	出力	入力	出力
----	----	----	----	----	----

Cmd	Index	Addr.	Len	Intp.	A出力	B出力	C出力	A入力	B入力	C入力
0c	05	0.0	08	0.0	0c00	0c00	0c00	0000	0000	0000

LEDの点滅パターン

つまり、FMMUは、これまで私のような制御系のエンジニアが、何週間もかけてガリガリと書いていたスレーブのメモリマップ設計仕様書を、ご主人さまとメイドの話し合いだけで、数秒程度の時間内に自動的に作り上げてしまう、ということです。

—— これはラクだ

としましたよ、本気で*。

*) 制御系のエンジニアには、このような動的な自動構築(ダイナミックオートコンフィギュレーション

ョン)を嫌う人は多いです。障害が発生した時に、その設計図が手元にないと、原因を特定するのが難しくなるからです。



スレーブのメモリマップ設計仕様書を、ご主人さまとメイドの話し合いだけで、数秒程度の時間内に自動的に作り上げてしまうのです

また、このFMMUのすごいところは、従来の通信では必ず必要となっていた、通信フレームのバッファリングと、通信フレームの中にある情報を探し回るという手間を、完全に一掃してしまったということです。

これが、EtherCATが、ゼロバッファで、超高速通信を実現している理由です(第2回「[EtherCAT通信の仕組みを知ろう～メイドは超一流のスナイパー!?](#)」参照)。

□

さて、前半はここまでと致します。

後半では、SyncManager(SM)と、実際にステップングモーターを動かすプログラムの作り方(simple_test.cの改造)の説明をしたいと思います。

来月、モーターを回してみたい方は、DI/DOのEtherCATスレーブとモーター“TD62003AP”を準備して、お待ちしております。

⇒[「EtherCATでホームセキュリティシステムを作る」連載バックナンバー](#)

特別協力:

本連載では、スレーブの提供などで [ベッコフオートメーション](#) にご協力いただいております。

BECKHOFF



Profile

江端智一(えばたともち)

日本の大手総合電機メーカーの主任研究員。1991年に入社。「サンマとサバ」を2種類のセンサーだけで判別するという電子レンジの食品自動判別アルゴリズムの発明を皮切りに、エンジン制御からネットワーク監視、無線ネットワーク、屋内GPS、鉄道システムまで幅広い分野の研究開発に携わる。

意外な視点から繰り出される特許発明には定評が高く、特許権に関して強いこだわりを持つ。特に熾烈(しれつ)を極めた海外特許庁との戦いにおいて、審査官を交代させるまで戦い抜いて特許査定を奪取した話は、今なお伝説として「本人」が語り継いでいる。共同研究のために赴任した米国での2年間の生活では、会話の1割の単語だけを拾って残りの9割を推測し、相手の言っている内容を理解しないで会話を強行するという希少な能力を獲得し、凱旋帰国。

私生活においては、辛辣(しんらつ)な切り口で語られるエッセイをWebサイト「[こぼれネット](#)」で発表し続け、カルト的なファンから圧倒的な支持を得ている。また週末には、LANを敷設するために自宅の庭に穴を掘り、侵入検知センサーを設置し、24時間体制のホームセキュリティシステムを構築することを趣味としている。このシステムは現在も拡張を続けており、その完成形態は「本人」も知らない。

本連載の内容は、個人の意見および見解であり、所属する組織を代表したものではありません。

関連記事



[海外出張に行くあなたは、「たった一人の軍隊」である](#)

われわれ英語に愛されないエンジニアが海外に出張する上で、「最大の難所」とも言える場所一つ。それが入国審査です。今回の実践編(入国・出国)では、海外出張に行くあなたを「たった一人の軍隊」とみなし、敵国(=出張先)に首尾よく潜入(=入国)する方法についてお話ししましょう。入国審査で使える“レジユメ”も紹介し

ます。



「海外で仕事をしたい」なんて一言も言っていない!

世の中にはいろいろな方がいますが、大きく2種類の人間に大別できます。「英語に愛される人間」と「英語に愛されない人間」です。



iPhoneも射程圏? IntelがCDMA技術を獲得

Intelはこのほど、CDMAモデムに関する技術資産を台湾Via Telecomから購入した。CDMA技術を新たに手に入れたことは、何を意味するのだろうか――。



1人の男が挑戦する“スマートホーム”の答え探し

マンションなどのリノベーションを行うリノベるは2015年9月、スマートハウスのショールーム「Connectly Lab.」をオープンした。世界中のスマートデバイスが集まる同ショールームで、開設したきっかけやその狙い、今後の展望を聞いた。そこには、ある1人の男の挑戦があった。

Copyright© 2016 ITmedia, Inc. All Rights Reserved.

